J Math Imaging Vis 26: 327–343, 2006

DOI: 10.1007/s10851-006-9003-7

The Minimal Structure and Motion Problems with Missing Data for 1D Retina Vision

MAGNUS OSKARSSON
Centre for Mathematical Sciences, Lund University, P O Box 118, SE-221 00 Lund Sweden
magnuso@maths.lth.se

KALLE ÅSTRÖM
Centre for Mathematical Sciences, Lund University, P O Box 118, SE-221 00 Lund Sweden
kalle@maths.lth.se

NIELS CHR. OVERGAARD
School of Technology and Society, Malmö University, SE-205 06 Malmö, Sweden
nco@ts.mah.se

Published online: 4 September 2006

Abstract. In this paper we investigate the structure and motion problem for calibrated one-dimensional projections of a two-dimensional environment. The theory of one-dimensional cameras are useful in several areas, e.g. within robotics, autonomous guided vehicles, projection of lines in ordinary vision and vision of vehicles undergoing so called planar motion. In a previous paper the structure and motion problem for all cases with non-missing data was classified and solved. Our aim is here to classify all structure and motion problems, even those with missing data, and to solve them. In the classification we introduce the notion of a prime problem. A prime problem is a minimal problem that does not contain a minimal problem as a sub-problem. We further show that there are infinitely many such prime problems. We give solutions to four prime problems, and using the duality of Carlsson these can be extended to solutions of seven prime problems. Finally we give some experimental results based on synthetic data.

1. Introduction

Understanding of one-dimensional cameras is important in several applications. In [18] it was shown that the structure and motion problem using line features in the special case of affine cameras can be reduced to the structure and motion problem for points in one dimension less, i.e. one-dimensional cameras. Thus solution to 1D structure and motion problems have been used to solve structure and motion problems for lines, [5, 18].

Another area of application is vision for planar motion. It is shown that ordinary vision (two-dimensional retina) can be reduced to that of one-dimensional cameras if the motion is planar, i.e. if the camera is rotating and translating in one specific plane only, cf. [11]. In another paper the planar motion is used for auto-calibration [1]. A typical example is the case where a camera is mounted on a vehicle or car that moves on a flat plane or flat road.

Our personal motivation, however, stems from the *autonomous guided vehicles*, called *AGV*, which are important components for factory automation. The navigation system uses strips of inexpensive reflector tape which are put on walls or objects along the route of the vehicle, cf. [14]. The *laser scanner* measures the direction from the vehicle to the beacons, but not the distance. This is the information used to calculate the position of the vehicle.

One of the primary vision problems (both 1D and 2D retina) is the so called structure and motion problem. For AGV's this is the procedure to obtain a map of the unknown positions of the beacons using images at unknown positions and orientations, cf. [2]. This is usually done off-line, once and for all, when the system is installed and then occasionally if there are changes in the environment. High-accuracy is needed, since the map has to be hard-coded in the system. The performance of the navigation routines depends on the precision of the reconstructed map. Note that the discussion here is focused on finding initial estimates of structure and motion. In practice it is necessary to refine these estimates using non-linear optimization or bundle adjustment, cf. [3, 19]. Minimal cases are also useful in robust estimation algorithms like RANSAC [20] for finding correspondences.

Part of this work has been presented in [4, 16, 17]. The overall goal of this work is to solve all solvable structure and motion problems. The purpose of this paper is twofold. Firstly, tools are developed to classify the minimal structure and motion problems with missing data. Secondly, solutions to the structure and motion problem are given for some of these minimal problems.

The paper is organized as follows. In Section 2 a background to the problem is given, describing the geometry of the problem. In Section 3 the structure and motion problem with missing data is formulated. In Section 4 the basis for classification of structure and motion problems with missing data is given, describing that the problems may belong to one or two of four different classes of problems. The actual classification is done in the following Section 5. Both algorithms for finding the number of members of the different classes and for actually finding out which class a structure and motion problem belongs to are given. In Section 6 we solve seven different prime problems. In Section 7 we give some experimental results based on synthetic data, and finally some conclusions are given in Section 8.

2. Background

A laser navigated vehicle is shown inFigure 1.a. The laser scanner, which is shown in detail in Figure 1.b, is mounted on the top of the vehicle. A laser beam gen-

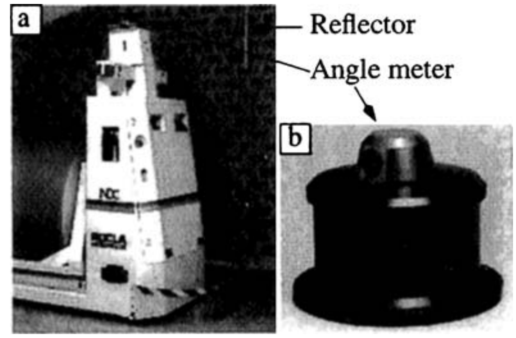

Figure 1. a: A laser guided vehicle. b: A laser scanner or angle meter.

erated by a vertical laser in the scanner, is deflected by a rotating mirror, at the top of the scanner. Thus, the laser beam scans the room at a fixed height. When the laser beam hits a beacon (a retroreflective tape, also shown in Figure 1.a), a large part of the light is reflected back to the scanner. The reflected light is processed to find sharp intensity changes. When this happens the bearing α of the laser beam relative to a fixed direction of the scanner is stored. All beacons are identical. This means that the identity of a beacon cannot be determined from a single measurement.

Introduce an object coordinate system which will be held fixed with respect to the scene. The bearing α defined above, depends on the position of the beacon (U_x, U_y) and of the position (P_x, P_y) and orientation P_θ of the scanner, cf. Figure 2,according to

$$\alpha(P, U) = \arg(U_x - P_x + i(U_y - P_y)) - P_\theta \,, \quad (1)$$

wherearg is the complex argument (the angle of the vector $(U_x - P_x, U_y - P_y)$ relative to the positive x-axis). The vector (P_x, P_y, P_θ) is called the *camera state*.

The above equation (1) for the measured bearing is non-linear. A somewhat simpler representation of the same equation can be obtained as follows. The vector between the camera center and the beacon can be written as

$$\lambda \begin{bmatrix} \cos(\alpha + P_\theta) \\ \sin(\alpha + P_\theta) \end{bmatrix} = \begin{bmatrix} U_x - P_x \\ U_y - P_y \end{bmatrix} = \begin{bmatrix} 1 & 0 & -P_x \\ 0 & 1 & -P_y \end{bmatrix} \begin{bmatrix} U_x \\ U_y \\ 1 \end{bmatrix} . \quad (2)$$

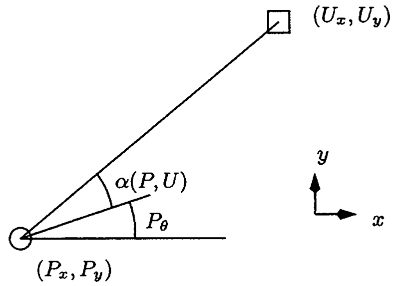

Figure 2. The Figure illustrates the measured angle α as a function of scanner position (P_x, P_y), scanner orientation P_θ and beacon position (U_x, U_y).

By multiplying each side with a rotation matrix we obtain

$$\lambda \underbrace{\begin{bmatrix} \cos(\alpha) \\ \sin(\alpha) \end{bmatrix}}_{\mathbf{u}} = \underbrace{\begin{bmatrix} \cos(P_\theta) & \sin(P_\theta) \\ -\sin(P_\theta) & \cos(P_\theta) \end{bmatrix} \begin{bmatrix} 1 & 0 & -P_x \\ 0 & 1 & -P_y \end{bmatrix}}_{\mathbf{P}} \underbrace{\begin{bmatrix} U_x \\ U_y \\ 1 \end{bmatrix}}_{\mathbf{U}}. \tag{3}$$

Thus the sensor follows the well known projection equation:

$$\lambda \mathbf{u} = \mathbf{P}\mathbf{U}. \tag{4}$$

Here the camera matrix is calibrated, i.e. it has the following form:

$$\mathbf{P} = \begin{bmatrix} a & b & c \\ -b & a & d \end{bmatrix}, \tag{5}$$

It is sometimes useful to consider dual image coordinates

$$\alpha \longleftrightarrow \mathbf{v} = \begin{bmatrix} -\sin(\alpha) & \cos(\alpha) \end{bmatrix}, \tag{6}$$

so that $\mathbf{vu} = 0$. This is particularly useful since it simplifies the camera constraint (4) to $\mathbf{vPU} = 0$. Notice that the constraint above is linear in the camera matrix. Denote by $\mathbf{p}$ the vector containing the four parameters of the camera matrix $\mathbf{p} = (a \quad b \quad c \quad d)^T$. Introduce the operator $D : \mathbb{R}^3 \to \mathbb{R}^{2\times 4}$ according to

$$D\left(\begin{pmatrix} X \\ Y \\ Z \end{pmatrix}\right) = \begin{pmatrix} X & Y & Z & 0 \\ Y & -X & 0 & Z \end{pmatrix}$$

Observe that $\mathbf{PU} = D(\mathbf{U})\mathbf{p}$. Thus it is possible to rearrange the equations so to obtain a linear constraint of the following type.

$$0 = \mathbf{vPU} = \mathbf{v}D(\mathbf{U})\mathbf{p}. \tag{7}$$

We will often use capital I to denote image number and capital J to denote point number. Thus $\mathbf{u}_{I,J}$ denotes the image direction for point J in image I, $\mathbf{P}_I$ denotes camera matrix for image I and $\mathbf{U}_J$ denotes object point number J.

3. Problem Formulation

Motivated by the previous sections the structure and motion problem will now be defined.

Problem 3.1 *Given some of the bearings to n beacons from m different positions* $\mathbf{u}_{I,J}$, $(I, J) \in \mathbb{I}$, *where* $\mathbb{I}$ *is an index set representing which beacons J are visible from image number I. The* **structure and motion problem** *is to find the depths* $\lambda_{I,J} > 0$, *the reconstructed points* $\mathbf{U}_J$ *and the camera matrices* $\mathbf{P}_I$ *such that*

$$\lambda_{I,J}\mathbf{u}_{I,J} = \mathbf{P}_I\mathbf{U}_J, \qquad \forall (I, J) \in \mathbb{I}.$$

We will in general let m denote the total number of images/cameras and n the total number of points, for a given problem. In this paper the interest lies in classifying and solving such problems. As such we will consider the problem with both beacons and cameras in general positions. As in ordinary vision there exist so-called critical configurations where there is an inherent ambiguity of the solutions to the structure and motion problem irrespective of the number cameras and points. In this paper we assume un-critical configurations. For non-missing data and 1D retina the issue of critical configurations was completely resolved in [6]. For missing data it is not known what the critical configurations are, but in order to understand which they are an understanding of the minimal cases for missing data is desirable.

The question whether a structure and motion problem is well-defined or perhaps even over-constrained depends on the structure of the index set $\mathbb{I}$.

In a previous paper [7] we considered only the cases where all beacons are visible in all views. The conclusion there is that the structure and motion problem is well-defined if and only if there are at least 3 views of at least 4 beacons, excluding the case $m = 3, n = 4$.

If it is possible to solve a case with a subset of cameras and beacons, then it is relatively easy to extend that solution to other cameras and points by well known techniques called resection and intersection, [7]. Thus the solution of any well-defined case above is based on the only two minimal cases with non-missing data, i.e. 4 views of 4 beacons and 3 views of 5 beacons.

The goal of this paper is to repeat this for the case of missing data. Depending on the index set $\mathbb{I}$ a structure and motion problem can be either

- ill-defined, if there is not, in general, enough data to constrain all unknown variables.
- well-defined and minimal, if there is exactly enough data to constrain the unknown variables (up to a discrete number of solutions).
- well-defined but over-constrained, if there is more than enough data to constrain the unknown variables.

One of the goals of this research is to classify the possible index sets $\mathbb{I}$ into these three categories, and if possible to design algorithms for solving the structure and motion problem in those cases where the problem is well defined.

Some of the minimal cases contain a minimal case as a subproblem. An example of this is the case with four points seen in five images, but where the fourth point is missing from the fifth image. It is minimal, but contains a subproblem (the problem with the first four views only) which is well defined and minimal. We will use the notation **prime problem** for a minimal problem which does not contain a well defined minimal problem as a subproblem. A minimal but not prime problem may in some cases be solved by first solving the contained prime problem and then extend the solution using resection and intersection. In other cases the prime problem may be embedded in the minimal problem in a more complicated manner. We first observe that similar to the case of non-missing data a well-defined but over-constrained problem contains as a subset a problem which is well-defined but minimal. Thus by finding the minimal cases and solving them, we should be able to solve all well-defined problems by the following algorithm:

1. Find whether a problem contains a well-defined minimal problem as a subset.
2. Solve the structure and motion problem for this subset.
3. Extend the solution to the original problem.

As the classification is based on the index set $\mathbb{I}$ alone, it is interesting to study these sets. In this paper we consider these sets as binary matrices, visibility matrices, A of size $m \times n$ where black denotes missing data and white denotes a measurement beacon which is present. Another way of viewing these index sets is as bi-partite graphs with $m + n$ nodes. There is an edge between node I in the first set and the node J in the second set if the point J is visible in image I. Thus a well-defined minimal case can be considered to be a sub-graph of a well-defined but over-constrained problem.

In the paper we will use the notation $|\mathbb{I}|$ to denote the number of elements in the set $\mathbb{I}$.

4. Classification of Structure and Motion Problems

The goal of this section is to give some conditions on what constitutes a well defined minimal problem. From these minimal problems the prime problems can be determined.

4.1. Equivalence Classes of Index Sets

The labeling of the cameras and of the beacons are of no consequence to the structure of the problem under study. Two index sets are considered equivalent if one results from the other by suitable relabelings. This means that there are many structure and motion problems that have different $\mathbb{I}$ but that correspond in principle to the same problem.

Definition 4.1. An index set $\mathbb{I}$ is said to be of type (m, n, l) if it represents a situation with m images and n points, in which exactly l points are not visible in all of the images, that is, if $|\mathbb{I}| = mn - l$.

From this definition it is clear that an index set $\mathbb{I}$ of type (m, n, l) can be represented by a binary $m \times n$–matrix $A = (a_{IJ})$ with $a_{IJ} = 1$ if $(I, J) \in \mathbb{I}$, and $a_{IJ} = 0$ otherwise, and such that $\sum_{IJ} a_{IJ} = mn - l$. The possible index sets of type (m, n, l) are thus in one–to–one correspondence with the set

$$M(m, n, l) = \{A \in \mathrm{Mat}_{m\times n}(\mathbb{Z}_2) : \textstyle\sum_{IJ} a_{IJ} = mn - l\}.$$

Let S_k denote the group of permutations on k symbols. With each permutation $\sigma \in S_k$ is associated a $k\times k$–permutation matrix $(\delta_{i\sigma(j)})$, which will be denoted simply by σ.

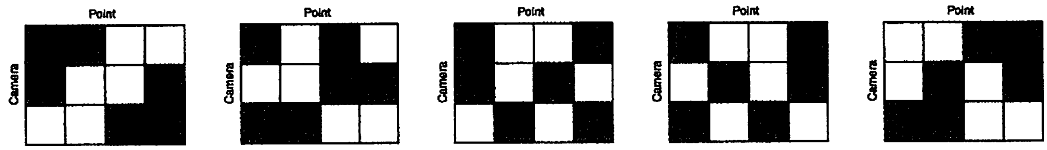

Figure 3. A number of permutation equivalent configurations

Definition 4.2. Two $m \times n$–matrices A and B are said to be *permutation equivalent*, if there exist permutations $\sigma \in S_m$ and $\tau \in S_n$ such that $B = \sigma^T A \tau$. If A and B are permutation equivalent then we write $A \sim B$.

The notion of equivalence of index sets can now be given a formal definition

Definition 4.3. Two index sets $\mathbb{I}$ and $\mathbb{I}'$ are called *equivalent*, and we write $\mathbb{I} \sim \mathbb{I}'$, if their corresponding matrix representations are permutation equivalent.

In Figure 3a number of permutation equivalent configurations is shown. The relation $\sim$ is easily seen to be an equivalence relation. It follows that $M(m, n, l)$ (or the corresponding index sets) can be partitioned into equivalence classes $M_1, \ldots, M_\omega$ of matrices (or index sets). The number of essentially different index sets is thus seen to be exactly the same as the number $\omega = \omega(m, n, l)$ of equivalence classes. This is the number of principally different problems of type (m, n, l).

4.2. *The Germs*

A first characterization of a well defined minimal structure and motions problem is that it contains exactly the same number of equations as unknowns. Each object point has two degrees of freedom and each camera state has three. The solution is only defined up to a similarity transformation. This manifold has dimension 4. Using n points and m cameras we thus have $2n + 3m - 4$ degrees of freedom in the parameters. Each measured bearing gives one constraint on the estimated parameters. Thus for a problem with visibility index set $\mathbb{I}$ we have $|\mathbb{I}|$ equations. This means that minimal problems have $|\mathbb{I}| = 2n + 3m - 4$. Since the maximum number of equations with m views of n points is mn it is easy to see how many measurements l that have to be occluded to obtain minimal problems, $l = mn - (2n + 3m - 4)$. This number is shown in Table 1.

In order to find the minimal problems we concentrate our efforts on problems of type $(m, n, mn - (2n + 3m - 4))$.

Definition 4.4. A structure and motion problem of type $(m, n, mn - (2n + 3m - 4))$ is said to be a *germ* of a minimal problem.

For a structure and motion problem to be minimal and/or prime the condition of being a germ is of course only a necessary condition.

4.3. *The Prime Condition*

For a given germ the corresponding structure and motion problem can be minimal or ill-posed. If it is minimal it may or may not be prime. The question of which class a germ belongs to can be categorized in terms of the graph of the index set. We will use the following intuitive assumption.

Conjecture 4.1. *For a given germ with index set $\mathbb{I}$, the corresponding structure and motion problem is minimal iff no sub-graph of $\mathbb{I}$ is over determined.*

An empirical method for determining whether a problem is minimal and well defined is to calculate the Jacobian of the bundle adjustment problem and study its singular values. We have used this technique to empirically check our conjecture.

It is clear that if a sub-graph of a germ with index set $\mathbb{I}$ is over determined then there has to be a part of the problem that is under determined and hence the whole problem is ill-posed.

Table 1. The number of excess constraints $l = mn - (2n + 3m - 4)$ for the structure and motion problem with m images of n points.

l	n					
m	4	5	6	7	8	9
3	-1	0	1	2	3	4
4	0	2	4	6	8	
5	1	4	7	10		
6	2	6	10			
7	3	8				
8	4					

Theorem 4.1. *Given a germ with index set $\mathbb{I}$; At least one sub-graph of $\mathbb{I}$ is over determined $\Rightarrow$ the corresponding structure and motion problem is ill-posed*

We will henceforth identify the class of minimal problems with those that fulfill conjecture 4.1. Under this assumption the notion of being a prime problem can be given the following formal definition,

Definition 4.5. A *prime problem* is a germ with index set $\mathbb{I}$ such that all strict sub-graphs of $\mathbb{I}$ are under determined.

A minimal problem which is not prime is an extension of a prime problem. The extended minimal problem can in many cases be solved by a succession of resections and intersections based on the solution to the prime case. In other cases the extension can be more complicated. In Figure 5a minimal problem of type (4,7,6) is shown. This can be shown to be an extension of a prime problem of type (4,6,4), which is shown in Figure 5b. If one has a solution to the prime problem the final point of the first problem can be found by intersection by the two first cameras.

Definition 4.6. An extension of type (m, n) is an extension with m extra cameras and n extra points of a prime problem.

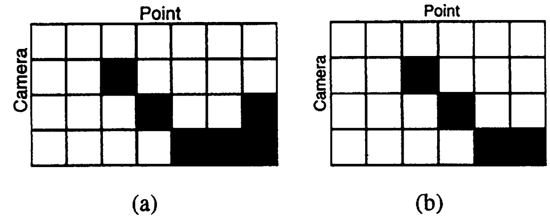

Figure 5. (a) A minimal problem of type (4,7,6) which is an extension of (b) a prime problem of type (4,6,4).

5. Finding and Classifying Germs

We now concentrate our efforts on finding out how many germs there are for different number of cameras and points. From these germs we then determine which are minimal and which are prime.

5.1. Equivalence Classes of Germs

Let the type (m, n, l) be fixed throughout the remainder of the discussion. To compute $\omega = \omega(m, n, l)$, notions and results from group theory will be used. Our reference here is to Section 3.6 of Fraleigh's text [12].

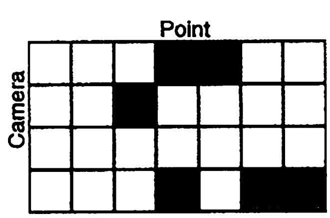

Figure 4. A germ of type (4,7,6).

First, denote the product group $S_m \times S_n$ by G. Secondly, if $g = (\sigma, \tau) \in G$ and $A \in M = M(m, n, l)$, then a group action of G on M is defined by the formula

$$g \cdot A = \sigma^T A \tau. \tag{8}$$

Thus two matrices $A, B \in M$ satisfy $A \sim B$ if and only if there exists $g \in G$ such that $g \cdot A = B$. The equivalence classes $M_1, \ldots, M_\omega$ of $\sim$ correspond to **the orbits in M under the action of G**. Therefore ω can be computed by the following well–known formula of Burnside; For any $g \in G$ let $M_g = \{A \in M : g \cdot A = A\}$ denote the set of matrices which are fix-points under action by g. Then

$$\omega = \frac{1}{|G|} \sum_{g \in G} |M_g|. \tag{9}$$

While (9) solves our problem in theory, there are still some practical problems to overcome. First, given $g \in G$, how do we compute $|M_g|$? Secondly, the sum $\sum_{g \in G} |M_g|$ must be evaluated, but as $|G| = m!n!$ becomes very large very quickly, the shear size of G may become an obstacle, unless the evaluation is performed cleverly.

A permutation $g = (\sigma, \tau) \in G$ may be regarded as an element of S_{mn}, as $A \mapsto \sigma^T A \tau$ permutes the mn entries of A. Let $g = g_1 g_2 \cdots g_s$ be the factorization in S_{mn} of g into a product of commuting (or disjoint) cyclic permutations. It is now easy to see that $A \in M_g$ if and only if, the entries in A, which equal zero, are arranged in such a manner, that any cycle g_i is either completely occupied by entries equal to zero, or contains no such entry at all. It follows that $|M_g|$ equals the number of ways in which l zeros can be allocated to $m \times n$ entries, such that the condition just described is satisfied. It is clear from this discussion that $|M_g|$ only depends on g's cycle structure (the number of cycles and their lengths).

Definition 5.1. If $\sigma \in S_k$ is a permutation in k symbols, let $n_i(\sigma)$, $i = 1, \ldots, k$, denote the number of i–cycles in the factorization of σ into commuting cycles. The *cycle index of* σ is the polynomial

$$P_\sigma(x_1, x_2, \ldots, x_k) = x_1^{n_1(\sigma)} x_2^{n_2(\sigma)} \cdots x_k^{n_k(\sigma)}. \tag{10}$$

If $H < S_k$ is a (sub–)group of permutations, then the *cycle index of* H is the polynomial

$$P_H(x_1, x_2, \ldots, x_k) = |H|^{-1} \sum_{h \in H} P_h(x_1, x_2, \ldots, x_k).$$

It follows from the theory developed in [22] that $|M_g| = (l!)^{-1}(d/dx)^l P_g(1 + x, 1 + x^2, \ldots, 1 + x^{mn})|_{x=0}$, for any $g \in G$. This formula solves the first of our two problems. Furthermore, it follows from Burnside's formula (9) that

$$\omega = \frac{1}{l!}\left(\frac{d}{dx}\right)^l P_G(1 + x, 1 + x^2, \ldots, 1 + x^{mn}) \Big|_{x=0} \tag{11}$$

It turns out that the cycle index P_H is reasonably easy to compute when H is all of S_k. Now, $G = S_m \times S_n$ is a proper subgroup of S_{mn}, so in view of (11) our second problem above becomes: How do we compute P_G when the cycle indices of S_m and S_n are known? Again the authors of [22] provide the answer; They introduce a new operation beside the usual addition and multiplication, denoted $*$, on the ring of polynomials in the infinitely many variables $x_1, x_2, x_3, \ldots$, and with rational coefficients. The "product" is associative, commutative and distributive over both + and; so it suffices to describe $*$ on monomial factors x_i^m and x_j^n, in which case

$$x_i^m * x_j^n = x_{[i,j]}^{imjn/[i,j]}, \tag{12}$$

where $[i, j]$ is the least common multiple of i and j. The authors of [22] then proceed to prove the following beautiful result, which we have used to compute P_G:

Theorem 5.1. (**Wei and Xu**). *If $H < S_m$ and $K < S_n$ are (sub–)groups, then $H \times K < S_{mn}$, and $P_{H\times K} = P_H * P_K$.*

Example The cycle index of S_3 is $\frac{1}{6}(x_1^3 + 3x_1x_2 + 2x_3)$ so if $G = S_3 \times S_3$ then

$$\begin{aligned} P_G &= \frac{1}{6}(x_1^3 + 3x_1x_2 + 2x_3) * \frac{1}{6}(x_1^3 + 3x_1x_2 + 2x_3) \\ &= \frac{1}{36}(x_1^9 + 6x_1^3x_2^3 + 9x_1x_2^4 + 12x_3x_6 + 8x_3^3) \end{aligned}$$

and it follows from (11) that $\omega(3, 3, 3) = 6$.

The procedure for calculating ω, described above, was implemented in Maple. Using this program we are able to compute ω for any given (m, n, l) and in particular for the germs. Table 2 contains ω for the first few types (m, n, l), with l given by Table 1.

5.2. *Finding germs*

When solving the different minimal cases it is important to have representatives for the different equivalence classes of one minimal case, in order to be able to determine which are prime, and also in the actual solving of the underlying structure and motion problem.

When finding the equivalence classes for different (m, n, l) it is enough to investigate configurations of type (n, n, n) from which other configuration easily can be derived. In the light of this we will concentrate our efforts on such configurations.

If one were to generate all index sets of type (n, n, n) there are $\binom{n^2}{n}$ such configurations. This is a very large number as n increases. If one has already calculated the equivalence classes for the configurations of type $(n-1, n-1, n-1)$ the number of possible candidates for (n, n, n) can be reduced substantially.

Lemma 5.1. *Given A and B two $n \times n$ matrices that are permutation equivalent; If C is given by the $(n+1) \times (n+1)$-matrix that is A extended one row and one column with ones, and where one non-zero element is set to zero, then there exists a matrix D that is given by the $(n + 1) \times (n + 1)$-matrix that is B extended one row and one column with ones, and where one non-zero element is set to zero, such that C and D are permutation equivalent.*

Proof: The proof is obvious. □

This leads to algorithm 5.1.

Algorithm 5.1 (Finding Possible Candidates for Equivalence Classes).

1. *Given a representative of each equivalence class of type $(n-1, n-1, n-1)$, X_i, $i = 1 \ldots N$.*
2. *For each X_i construct Y_i, $i = 1 \ldots N$ where Y_i is the $n \times n$-matrix that is X_i extended one row and one column with ones.*

3. *In Y_i there are $n^2 - (n-1)$ non-zero elements. Y_{ij} is given by setting the j-th nonzero element of Y_i to zero.*

This gives a number of possible candidates from which representatives of the equivalence classes can be selected. Using algorithm 5.1 the number of possible candidates has been reduced from $\binom{n^2}{n}$ to $(n^2-n+1)\cdot N$. For instance if $n = 10$, then $N = 1430$ and this leads to a reduction from $1.7 \cdot 10^{13}$ to $1.3 \cdot 10^5$.

Short of trying all permutations there is no easy way of establishing if two index sets are permutation equivalent. If one is to check all possible permutations of a $n \times n$ matrix there are $(n!)^2$ such possibilities. This is rather undoable even for moderate n. In this section we propose an algorithm that uses a different approach.

If we have a set of matrices we try to permute each matrix to a standard form. This is done by solving the minimization problem stated in (13) for each Y in our set of matrices.

$$\hat{X} = \arg\min_{X\sim Y} f(X). \tag{13}$$

Here f is given by

$$f(X) = \sum_{i,j} 2^{a_{ij}} x_{ij}, \quad a_{ij} = n(i-1) + j - 1, \tag{14}$$

and x_{ij} are the entries in X. For each $n \times n$-matrix, with zero and one entries, f is injective since $f(X)$ represents the binary number with the n^2 entries of X as its digits. This means that $\hat{X}$ in (13) exists uniquely for every Y.

This leads to the following algorithm;

Algorithm 5.2 (Comparing matrices).

1. *Given a number of matrices X_i, $i = 1 \ldots N$,*
2. *Find $\hat{X}_i$ by solving $\hat{X}_i = \arg\min_{X\sim X_i} f(X)$.*
3. *$X_i \sim X_j$ if $\hat{X}_i = \hat{X}_j$.*

The problematic part of algorithm 5.2 is finding the global minimum. We have used a local search method when we do the minimization, and inevitably we end up with not finding the global minimum for every matrix in our list. This means that in most cases we will have to do something extra in order to find the distinct equivalence classes. One way of getting away from a local minimum is to start the minimization again from a position away from the starting position. This can be done by randomly permuting the given matrix and then use the minimization routines on the new matrix. The steps are described in algorithm 5.3.

Algorithm 5.3 (Comparing matrices again).

1. *Given a number of matrices X_i, $i = 1 \ldots N$*
2. *For each X_i create M random permutations X_{ij}, $j = 1 \ldots M$ of X_i.*
3. *For every X_{ij} find $\hat{X}_{ij}$ by solving $\hat{X}_{ij} = \arg\operatorname{loc}\min_{X\sim X_{ij}} f(X)$.*
4. *$X_i \sim X_k$ if $\hat{X}_{ij} = \hat{X}_{kl}$ for some j and l.*

Since we for every given configuration (m, n, l) can calculate the number of equivalence classes according to section 5.1 we can use algorithm 5.3 until we have found the right number of distinct matrices.

The motivation for the success of algorithm 5.3 is the following; Let's say that we have two matrices that are permutation equivalent, but that have ended up in different minima after the use of algorithm 5.2, and for simplicity assume that these are the only two minima that we may end up with, from different starting points, after our minimization. If minimum one attracts a of all starting matrices and minimum two attracts $(1-a)$ then the probability that the two matrices stay in their respective minima after the use of algorithm 5.3 is $a^M(1-a)^M$. This is maximized for $a = 0.5$ with probability equal to 2^{-2M} which tends rapidly towards zero as M increases.

Due to the fact that $\omega(l+a, l+b, l) = \omega(l, l, l)\ \forall a, b \in \mathbb{Z}_+$ sought germs of type (m, n, l) with m and n larger than l correspond to germs of type (l, l, l). Germs of type (m, n, l) with m or n smaller than l can be chosen from the equivalence classes for (l, l, l) where configurations that can not be contained in a $m \times n$ matrix have been removed.

We have calculated the equivalence classes for some of the first germs using the algorithms described in this section. In Table 2 the number of distinct germs for these cases are given.

5.3. *Classifying Germs*

From a number of germs the object is to classify them first of all as minimal or ill-defined. Then from the minimal germs we determine which are prime. A first condition that roots out many ill-defined germs is that to every camera there has to be at least three measurements in total otherwise it is impossible to determine

Table 2. The number ω of different germs for different m and n.

ω	n					
m	4	5	6	7	8	9
3	–	1	1	3	6	11
4	1	3	16	62	225	
5	1	16	155	1402		
6	3	79	1799			
7	6	361				
8	16					

Table 3. The number of minimal configurations for different m and n.

	n					
m	4	5	6	7	8	9
3	–	1	1	2	3	4
4	1	3	12	41	118	
5	1	12	110	876		
6	2	48	1050			
7	3	159				
8	5					

the camera uniquely. Similarly there has to be at least two measurements to every point in order to uniquely be able to determine the position of the point. From the germs that fulfill these criteria we try to evaluate which are minimal. The reason that some germs are ill-posed is that there are sub-problems that are over determined and sub-problems that are under determined but in total the number of measurements match the number of unknowns. This was described in section 4.3. So one way of determining which of the germs are minimal is to for every sub-graph of the given $\mathbb{I}$ check whether the sub-graph represents a well posed problem or not. For a graph with m cameras and n points there are 2^{m+n} sub-graphs so for larger m and n it may take a while to check all sub-graphs. Another way is to use the following algorithm:

Algorithm 5.4 (Classifying germs)

1. *Given the matrix representation A of an index set $\mathbb{I}$ of type (m, n, l), for all $3 \leq \widetilde{m} \leq m$, and for all possible ways choose $\widetilde{m}$ rows from A.*
2. *From the $\widetilde{m}$ rows of A choose the $\tilde{n}$ points that are seen in at least 3 of the $\widetilde{m}$ chosen cameras.*
3. *The total number of measurements in the chosen $\widetilde{m}$ cameras and $\tilde{n}$ points is N. If $N > 3\widetilde{m} + 2\tilde{n} - 4$ then $\mathbb{I}$ is ill-defined. If $N = 3\widetilde{m} + 2\tilde{n} - 4$ and $\widetilde{m} < m$ or $\tilde{n} < n$ then $\mathbb{I}$ is not prime.*

Using algorithm 5.4 we have calculated the number of minimal and prime configurations for some different values of m and n. The results are shown in Table 3, where the number of minimal configurations are shown, and in Table 4, wherethe number of prime configurations are shown.

In Figure 7 the prime problems for the configurations of type (5, 5, 4) and (4, 6, 4) are given. The configurations in Figure 7a–c seem to be connected to configurations in Figure 7d–f. The similarity can be explained using a technique that Carlsson developed in [8, 9].

Theorem 5.2. *The calibrated structure and motion problem with n points and m images is equivalent to the calibrated structure and motion problem with $(m + 1)$ points and $(n - 1)$ images.*

The theorem is based on singling out one special point, which must be visible in all views. By using this point as a reference point (origo). It is possible to exchange the cameras for beacons and vice versa. Thus a structure and motion problem with visibility matrix $[A_{m\times n-1}\, \mathbf{0}_{m\times 1}]$ is dual to one with visibility matrix $[A^T_{n-1\times m}\, \mathbf{0}_{n-1\times 1}]$. Using this duality one can see that the configurations in Figure 7a–c are dual to the configurations in Figure 7d–f. If one has the solution to one structure and motion problem the solution to its dual problem can easily be calculated.

The transformation between cameras and beacons in the two problems are given by a special Cremona transformation, described in the following lemma:

Lemma 5.2. *Consider the calibrated Cremona transformation*

$$(u, v, w) \mapsto (uw, -vw, (u^2 + v^2)) \ . \qquad (15)$$

The transformation has the property that from every point A the angle measured to an arbitrary point B

Table 4. The number of prime configurations for different m and n.

	n					
m	4	5	6	7	8	9
3	–	1	0	0	0	0
4	1	1	3	5	8	
5	0	3	22	145		
6	0	6	136			
7	0	0				
8	0					

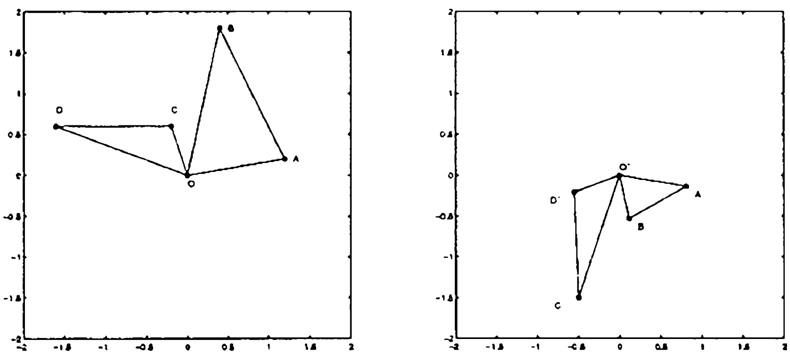

Figure 6. Example of Calibrated Cremona transformation.

relative to the origin is the same as the angle from the dual point B' to the point A'.

The lemma is illustrated by Figure 6.Notice that the triangle OAB is congruent to $OB'A'$, the triangles OCD is congruent to $OD'C'$ and similarly for any triangle with O as one of the vertices. This means that if we measure bearings from any set of camera positions $(C_1, \ldots, C_m)$ to the points $(X_1, \ldots, X_n)$ we will get the same angles as we would if we measured from the points $(X'_1, \ldots, X'_n)$ to the points $(C'_1, \ldots, C'_m)$.

If one looks at Table 4 the number of prime configurations seem to increase quickly as both m and n increase. This leads to the question whether this is true or if the number of prime cases after some time stop growing. One can at least give the following result,

Theorem 5.3. *There are infinitely many prime configurations.*

Proof: Given a germ of type $(m, m, m^2 - 5m + 4)$ one can construct the following prime configuration; the first point is seen in all images. The remaining $m - 1$ points are seen in exactly 4 images each. Of these $m - 1$ points, the first 4 cameras see exactly 3 points and the remaining $m - 4$ cameras see exactly 4 points. The construction is illustrated in Figure 8. We will use algorithm 5.4 to show that this construction is prime. For $\widetilde{m} \leq m - 2$: In order to use as much information as possible one should choose the $\widetilde{m}$ cameras close together. This gives in the best case $3\widetilde{m} + 2\widetilde{m} - 4 = 5\widetilde{m} - 4$ unknowns and $\widetilde{m} + 4(\widetilde{m} - 3) + 3 \cdot 2 = 5\widetilde{m} - 6$ constraints, so in this case it is always under-determined. For $\widetilde{m} = m - 1$ the same

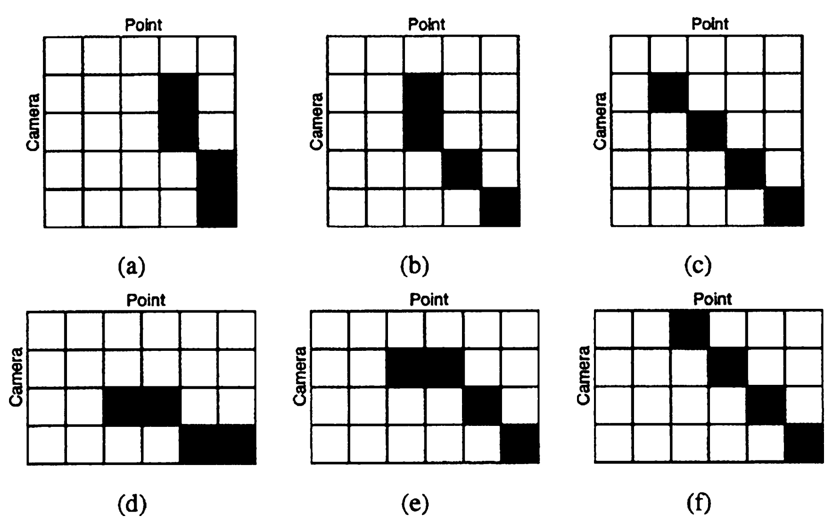

Figure 7. The three distinct configurations for prime cases of type (5, 5, 4), (a–c), and the three distinct configurations for prime cases of type (4, 6, 4), (d–f).

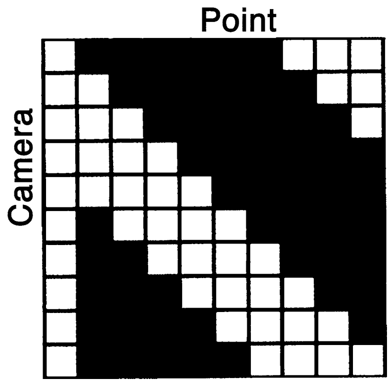

Figure 8. A prime configuration of type (10, 10, 54).

reasoning gives at best: $3\widetilde{m} + 2(\widetilde{m} + 1) - 4 = 5\widetilde{m} - 2$ unknowns and $\widetilde{m} + 4(\widetilde{m} - 3) + 3 \cdot 3 = 5\widetilde{m} - 3$ constraints. So also in this case it is always under-determined. Finally for $\widetilde{m} = m$ we have $3\widetilde{m} + 2\widetilde{m} - 4 = 5\widetilde{m} - 4$ unknowns matching the $\widetilde{m} + 4(\widetilde{m} - 1) = 5\widetilde{m} - 4$ constraints. □

Comparing Table 3 with Table 4 we see that there is only one prime case for four points. Similarly there is only one prime case for three cameras. The extensions in these cases are of type $(m, 0)$ and $(0, n)$. These type of extensions can always be solved using resection and intersection respectively. Extensions of type $(1, n)$ and $(m, 1)$ can always also be solved using only combinations of resection and intersection. The first more complicated extension occurs for the type (2, 2). In order for the extension not to be able to be solved with intersection and resection all cameras and points must be under-determined with respect to the prime configuration. And all cameras and points should be exactly determined with the information contained in the remaining four measurements. For an extension of type (2, 2) this can essentially only be done in one way. This extension is shown in Figure 9b. Other minimal extensions of type (2,2) can either not be solved or can be solved using resection and intersection. For example, consider Figure 9a with $(m + 2)$ cameras and $(n + 2)$ points. Point $(n + 2)$ can be intersected using cameras $(m - 1)$ and m. Camera $(m + 1)$ can then be estimated with resection using the n first points and point $(n + 2)$. Point $(n + 1)$ is then given by cameras m and $(m + 1)$. This finally gives camera $(m + 2)$ by resection of points $\{(n - 1) \ldots (n + 1)\}$.

6. Solution of Some Minimal Cases

In section 5 we determined exactly which the prime problems are. In this section we turn our attention to the task of solving some of these prime problems. We also saw that some minimal problems can be solved by extending a prime problem. One such case of an add-on is also considered in this section.

6.1. The Case of Five Points in Four Images

There is only one prime configuration for the case of five points in four images. This is the case where one sees five points in two images. In image three, one point is occluded and in image four another point is occluded. We will start by finding the solutions to this case.

Theorem 6.1. *The structure and motion problem with four views of five points*

$$\lambda_{I,J} \mathbf{u}_{I,J} = \mathbf{P}_I \mathbf{U}_J, \qquad \forall (I, J) \in \mathbb{I}.$$

with $\mathbb{I}$ such that point 1 is missing in view 3 and point 2

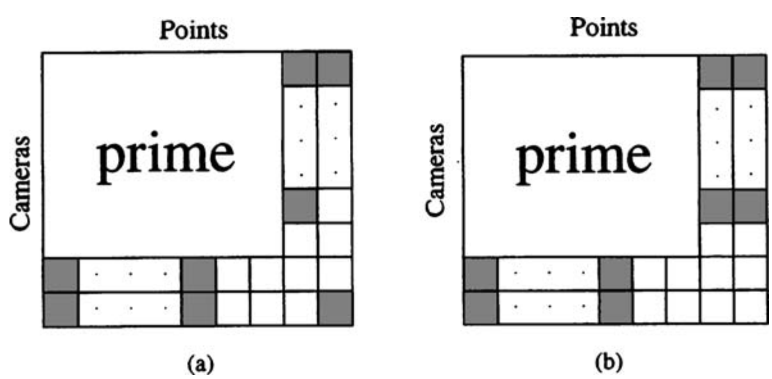

Figure 9. Extensions of type (2,2) from prime (m, n) problems to minimal $(m + 2, n + 2)$. In (a) the problem can be solved by a succession of resections and intersections. In (b) the only extension of type (2,2) not directly solvable by intersections and resections is shown.

is missing in view 4 (see Table **??***) has in general three solutions.*

Proof: We introduce a coordinate system such that the first camera is given by

$$\mathbf{P}_1 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}.$$

Then we can parameterize the structure with the depths in the first image,

$$\mathbf{U}_J = \begin{bmatrix} \lambda_{1,J}\cos(\alpha_{1,J}) & \lambda_{1,J}\sin(\alpha_{1,J}) & 1 \end{bmatrix}^T.$$

Using equation (7) we can write the projections in the remaining three images as

$$\mathbf{v}_{I,J} \begin{bmatrix} \lambda_{1,J}\cos(\alpha_{1,J}) & \lambda_{1,J}\sin(\alpha_{1,J}) & 1 & 0 \\ \lambda_{1,J}\sin(\alpha_{1,J}) & -\lambda_{1,J}\cos(\alpha_{1,J}) & 0 & 1 \end{bmatrix} \mathbf{p}_I = \mathbf{0}$$

If we write this as

$$M^2_{5\times 4}\mathbf{p}_2 = \mathbf{0},\ M^3_{4\times 4}\mathbf{p}_3 = \mathbf{0},\ M^4_{4\times 4}\mathbf{p}_4 = \mathbf{0}$$

we see that all 4×4-determinants of M^i have to be zero since $\mathbf{p_i} \neq \mathbf{0}$. This leads to seven two-degree polynomials in λ_{1i}. All seven are not linearly independent. Points are only determined up to scale so we can set $\lambda_{15} = 1$ in our calculations and then choose four of the polynomials and solve for λ_{1i}. If

$$\begin{aligned} p_{11} &= a\lambda_{12}\lambda_{13} + b\lambda_{13}\lambda_{14} + c\lambda_{14} + d\lambda_{12}\lambda_{14} + e\lambda_{13} + f\lambda_{12} \\ p_{12} &= g\lambda_{12}\lambda_{13} + h\lambda_{13}\lambda_{14} + i\lambda_{14} + j\lambda_{12}\lambda_{14} + k\lambda_{13} + l\lambda_{12} \\ p_{13} &= m\lambda_{11}\lambda_{13} + n\lambda_{13}\lambda_{14} + o\lambda_{14} + p\lambda_{11}\lambda_{14} + q\lambda_{13} + r\lambda_{11} \\ p_{14} &= s\lambda_{11}\lambda_{13} + t\lambda_{13}\lambda_{14} + u\lambda_{14} + v\lambda_{11}\lambda_{14} + w\lambda_{13} + x\lambda_{11}, \end{aligned}$$

then we want to solve the equations

$$p_{1i} = 0, \quad i = 1 \ldots 4. \tag{16}$$

Taking the resultant, cf. [10], of p_{11} and p_{12} with respect to λ_{12} and of p_{13} and p_{14} with respect to λ_{11} gives respectively polynomials p_{21} and p_{22} of total degree three,

$$\begin{aligned} p_{21} &= a'\lambda_{13}^2\lambda_{14} + b'\lambda_{13}\lambda_{14}^2 + c'\lambda_{13}^2 + d'\lambda_{14}^2 + e'\lambda_{13}\lambda_{14} + f'\lambda_{13} + g'\lambda_{14} \\ p_{22} &= h'\lambda_{13}^2\lambda_{14} + i'\lambda_{13}\lambda_{14}^2 + j'\lambda_{13}^2 + k'\lambda_{14}^2 + l'\lambda_{13}\lambda_{14} + m'\lambda_{13} + n'\lambda_{14}. \end{aligned}$$

Taking the resultant of p_{21} and p_{22} with respect to λ_{13} gives a homogeneous polynomial in λ_{14} of degree seven. One of the seven solutions is created from the resultant calculations, and will not fulfill the original polynomial equations. This gives six solutions for $\lambda_{1i}, i = 1 \ldots 5$ including $\lambda_{1i} = 0, i = 1 \ldots 5$. Of the five non-zero solutions, two are complex and are a result of our parameterization,

$$(\lambda_{11}, \lambda_{12}, \lambda_{13}\ , \lambda_{14}, \lambda_{15}) \sim (e^{\pm i\alpha_{11}}, e^{\pm i\alpha_{12}}, e^{\pm i\alpha_{13}}, e^{\pm i\alpha_{14}}, e^{\pm i\alpha_{15}}).$$

This leaves three non-trivial solutions. □

6.2. The Case of Five Points in Five Images

There are three prime problems for the case of five points in five images. We will now solve these three prime problems and their three dual cases.

Theorem 6.2. *The structure and motion problem for five images of five points,*

$$\lambda_{I,J}\mathbf{u}_{I,J} = \mathbf{P}_I\mathbf{U}_J, \qquad \forall (I, J) \in \mathbb{I}.$$

with $\mathbb{I}$ *given by Figure 7a has in general three solutions.*

Proof: As in the case with four images of five points we can use the first image to parameterize the structure. The remaining four images are then used to solve the problem. This gives four second degree polynomial equations in $\lambda_{1i}, i = 1 \ldots 4$. These polynomials have the exact same structure as those in the case of five points in four images, and hence the solutions have the same structure. This leads to that the problem of five points in five views has three non-trivial solutions. □

The dual to this case of five points in five images is the case of six points in four images given by Figure 7d. This means that there are three solutions to this case of six points in four images.

Corollary 6.1. *The structure and motion problem for four images of six points,*

$$\lambda_{I,J}\mathbf{u}_{I,J} = \mathbf{P}_I\mathbf{U}_J, \qquad \forall (I, J) \in \mathbb{I}.$$

with $\mathbb{I}$ *given by Figure 7d has in general three solutions.*

Using the same kind of parameterization as in the previous cases, we can solve the prime problem given by Figure 7b.

Theorem 6.3. *The structure and motion problem for five images of five points,*

$$\lambda_{I,J}\mathbf{u}_{I,J} = \mathbf{P}_I\mathbf{U}_J, \qquad \forall (I, J) \in \mathbb{I}.$$

with $\mathbb{I}$ given by Figure 7b has in general four solutions.

Proof: We parameterize the structure with image one, which leads to the following system of equations:

$$
\begin{aligned}
p_{11} &\equiv a\lambda_{12}\lambda_{13} + b\lambda_{13}\lambda_{14} + c\lambda_{14} + d\lambda_{12}\lambda_{14} + e\lambda_{13} + f\lambda_{12} = 0\\
p_{12} &\equiv g\lambda_{12}\lambda_{13} + h\lambda_{13}\lambda_{14} + i\lambda_{14} + j\lambda_{12}\lambda_{14} + k\lambda_{13} + l\lambda_{12} = 0\\
p_{13} &\equiv m\lambda_{11}\lambda_{13} + n\lambda_{13}\lambda_{14} + o\lambda_{14} + p\lambda_{11}\lambda_{14} + q\lambda_{13} + r\lambda_{11} = 0\\
p_{14} &\equiv s\lambda_{11}\lambda_{12} + t\lambda_{12}\lambda_{14} + u\lambda_{14} + v\lambda_{11}\lambda_{14} + w\lambda_{12} + x\lambda_{11} = 0.
\end{aligned}
$$

Taking the resultant of p_{13} and p_{14} with respect to λ_{11} gives a polynomial in λ_{12}, λ_{13} and λ_{14}. Taking the resultant of this polynomial and p_{11} with respect to λ_{12} gives the following polynomial of degree four in λ_{13} and λ_{14},

$$
p_{22} = h'\lambda_{13}\lambda_{14}^3 + i'\lambda_{14} + j'\lambda_{13} + k'\lambda_{14}^2 + l'\lambda_{13}\lambda_{14} + m'\lambda_{13}^2\lambda_{14} + n'\lambda_{13}^2 + o'\lambda_{14}^3 + p'\lambda_{13}^2\lambda_{14}^2.
$$

Taking the resultant of p_{11} and p_{12} with respect to λ_{12} gives the following polynomial in λ_{13} and λ_{14},

$$
p_{21} = a'\lambda_{13}^2\lambda_{14} + b'\lambda_{13}\lambda_{14}^2 + c'\lambda_{13}^2 + d'\lambda_{14}^2 + e'\lambda_{13}\lambda_{14} + f'\lambda_{13} + g'\lambda_{14}.
$$

Taking the resultant of p_{21} and p_{22} with respect to λ_{13} gives a homogeneous polynomial in λ_{14} of degree nine. Two of the nine solutions are created from the resultant calculations, and will not fulfill the original polynomial equations. This gives seven solutions for $\lambda_{1i}, i = 1\ldots5$ including the trivial one. Of these, two are complex and are a result of our parameterization,

$$
(\lambda_{11}, \lambda_{12}, \lambda_{13}, \lambda_{14}, \lambda_{15}) \sim (e^{\pm i\alpha_{11}}, e^{\pm i\alpha_{12}}, e^{\pm i\alpha_{13}}, e^{\pm i\alpha_{14}}, e^{\pm i\alpha_{15}}).
$$

This leaves four non-trivial solutions. □

The dual to this case of five points in five images is the case of six points in four images given by Figure 7e. This means that there are three solutions to this case of six points in four images.

Corollary 6.2. *The structure and motion problem for four images of six points,*

$$
\lambda_{I,J}\mathbf{u}_{I,J} = \mathbf{P}_I\mathbf{U}_J, \qquad \forall(I,J) \in \mathbb{I}.
$$

with $\mathbb{I}$ given by Figure 7e has in general four solutions.

Finally the last prime case of five images of five points can be shown to have five solutions.

Theorem 6.4. *The structure and motion problem for five images of five points,*

$$
\lambda_{I,J}\mathbf{u}_{I,J} = \mathbf{P}_I\mathbf{U}_J, \qquad \forall(I,J) \in \mathbb{I}.
$$

with $\mathbb{I}$ given by Figure 7c has in general five solutions.

Proof: We parameterize the structure with image one, which leads to the following system of equations:

$$
\begin{aligned}
p_{11} &\equiv a\lambda_{12}\lambda_{13} + b\lambda_{13}\lambda_{14} + c\lambda_{14} + d\lambda_{12}\lambda_{14} + e\lambda_{13} + f\lambda_{12} = 0\\
p_{12} &\equiv g\lambda_{12}\lambda_{13} + h\lambda_{13}\lambda_{14} + i\lambda_{14} + j\lambda_{12}\lambda_{14} + k\lambda_{13} + l\lambda_{12} = 0\\
p_{13} &\equiv m\lambda_{11}\lambda_{13} + n\lambda_{13}\lambda_{14} + o\lambda_{14} + p\lambda_{11}\lambda_{14} + q\lambda_{13} + r\lambda_{11} = 0\\
p_{14} &\equiv s\lambda_{11}\lambda_{13} + t\lambda_{13}\lambda_{14} + u\lambda_{14} + v\lambda_{11}\lambda_{14} + w\lambda_{13} + x\lambda_{11} = 0.
\end{aligned}
$$

Taking the resultant of p_{12} and p_{13} with respect to λ_{11} and the resultant of p_{12} and p_{14} with respect to λ_{11} gives respectively the following polynomials in λ_{12}, λ_{13} and λ_{14} :

$$
\begin{aligned}
p_{21} &= a'\lambda_{12}\lambda_{13} + b'\lambda_{13}\lambda_{14} + c'\lambda_{12}\lambda_{14} + d'\lambda_{12} + e'\lambda_{13} + f'\lambda_{14}\\
&\quad + g'\lambda_{14}^2 + h'\lambda_{12}\lambda_{13}\lambda_{14} + i'\lambda_{13}\lambda_{14}^2 + j'\lambda_{12}\lambda_{14}^2\\
p_{22} &= k'\lambda_{12}\lambda_{14} + l'\lambda_{14}\lambda_{13} + m'\lambda_{12}\lambda_{13} + n'\lambda_{12} + o'\lambda_{14} + p'\lambda_{13}\\
&\quad + q'\lambda_{13}^2 + r'\lambda_{12}\lambda_{14}\lambda_{13} + s'\lambda_{14}\lambda_{13}^2 + t'\lambda_{12}\lambda_{13}^2.
\end{aligned}
$$

Taking the resultants of p_{11} and p_{21} and of p_{11} and p_{22} with respect to λ_{12} gives the following two polynomials in λ_{13} and λ_{14} :

$$
\begin{aligned}
p_{31} &= a''\lambda_{13} + b''\lambda_{14} + c''\lambda_{13}\lambda_{14} + d''\lambda_{13}^2 + e''\lambda_{14}^2\\
&\quad + f''\lambda_{13}\lambda_{14}^2 + g''\lambda_{13}^2\lambda_{14} + h''\lambda_{14}^3 + i''\lambda_{13}^2\lambda_{14}^2 + j''\lambda_{13}\lambda_{14}^3\\
p_{32} &= k''\lambda_{13} + l''\lambda_{14} + m''\lambda_{13}\lambda_{14} + n''\lambda_{13}^2 + o''\lambda_{14}^2\\
&\quad + p''\lambda_{13}\lambda_{14}^2 + q''\lambda_{13}^2\lambda_{14} + r''\lambda_{13}^3 + s''\lambda_{13}^2\lambda_{14}^2 + t''\lambda_{13}^3\lambda_{14}.
\end{aligned}
$$

Taking the resultant of p_{31} and p_{32} with respect to λ_{13} gives a homogeneous polynomial in λ_{14} of degree twelve. Four of the twelve solutions are created from the resultant calculations, and will not fulfill the original polynomial equations. This leaves eight solutions for $\lambda_{1i}, i = 1\ldots5$ including the trivial one. Of these, two are complex and are a result of our parameterization,

$$
(\lambda_{11}, \lambda_{12}, \lambda_{13}, \lambda_{14}, \lambda_{15}) \sim (e^{\pm i\alpha_{11}}, e^{\pm i\alpha_{12}}, e^{\pm i\alpha_{13}}, e^{\pm i\alpha_{14}}, e^{\pm i\alpha_{15}}).
$$

This leaves five non-trivial solutions. □

Table 5. Some bearing measurements

0.6929	−0.7825	−1.9347	0.3263	−0.6421
0.3206	−0.9479	−1.8732	−0.0041	−0.8289
–	−2.5202	2.4474	−0.9746	−2.3323
2.3024	–	−1.0540	1.8991	0.6499

The dual to this case of five points in five images is the case of six points in four images given by Figure 7f. This means that there are five solutions to this case of six points in four images.

Corollary 6.3. *The structure and motion problem for four images of six points,*

$$\lambda_{I,J}\mathbf{u}_{I,J} = \mathbf{P}_I\mathbf{U}_J, \qquad \forall(I,J) \in \mathbb{I}.$$

with $\mathbb{I}$ given by Figure 7f has in general five solutions.

6.3. The Two-by-Two Extension

In section 5.3 we saw that apart from simple extensions based on intersection and resection the first extension of a prime problem is the extension by two cameras and two points. This type of extension is shown in Figure 9b. We assume that we have a solution to a prime problem with m cameras and n points. The task is then to extend this solution to the solution of the extended problem with $m+2$ cameras and $n+2$ points. Two extra cameras and two extra points means that we have $2\cdot 3 + 2\cdot 2 = 10$ unknowns to solve for. Using intersection, the known cameras give 2 linear constraints on the unknown points. And using resection the known points give 4 linear constraints on the unknown cameras. This leaves 4 parameters, one for each camera ($A_i,\ i=1,2$) and one for each point ($X_i,\ i=1,2$), to solve for. The two new points are seen in both the two new views. This gives four quadratic constraints on the four parameters,

$$a_{ij}X_iA_j + b_{ij}X_i + c_{ij}A_j + d_{ij} = 0, \quad i=1,2, \quad j=1,2$$

with the coefficients $(a_{ij}, b_{ij}, c_{ij}, d_{ij})$ only depending on the images. This means that there could be up to $2^4 = 16$ solutions according to Bezout's theorem. But due to the sparseness of the polynomials this is not the case. Taking resultants pairwise we can eliminate X_i. This leaves two polynomial equations in A_i,

$$a'_iA_1A_2 + b'_iA_1 + c'_iA_2 + d'_i = 0, \quad i = 1,2.$$

Taking the resultant of the two polynomials with respect to A_2 leaves the following quadratic equation in A_1

$$a''A_1^2 + b''A_1 + c'' = 0.$$

This discussion leads to the following result:

Theorem 6.5. *Given an extension of type* $(2,2)$ *to a structure and motion problem with m cameras and n points (as depicted in Figure 9b), the number of solutions are in general $2\times N$, where N is the number of solutions of the original problem with m cameras and n points.*

7. Some Experimental Results

The methods described in the proof of theorem 6.1 can easily be implemented. In Table 5 bearings for an example of the minimal case described in theorem 6.1 is shown. The resulting solutions are given in Figure 10. In this case there were three real solutions with all depths positive.

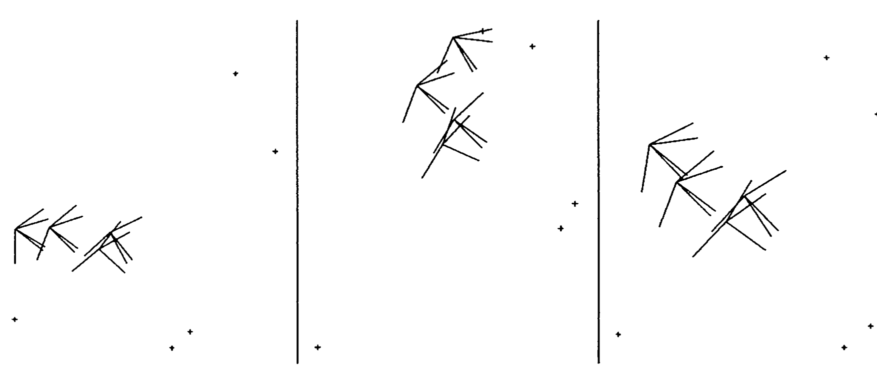

Figure 10. Three solutions to the minimal case of five points in four images. Beacons are indicated by '+'.

Table 6. Some bearing measurements

−1.9786	−0.5736	0.8046	−1.0507	0.5931	–	–
−2.5202	−1.1710	1.5796	−1.8684	1.2136	–	–
−0.8188	0.6134	3.0339	-0.0885	2.6759	−0.1730	−2.4311
−2.2663	−0.6898	–	–	–	−1.3617	1.5151
−1.8792	−0.3047	–	–	–	−1.1115	2.5170

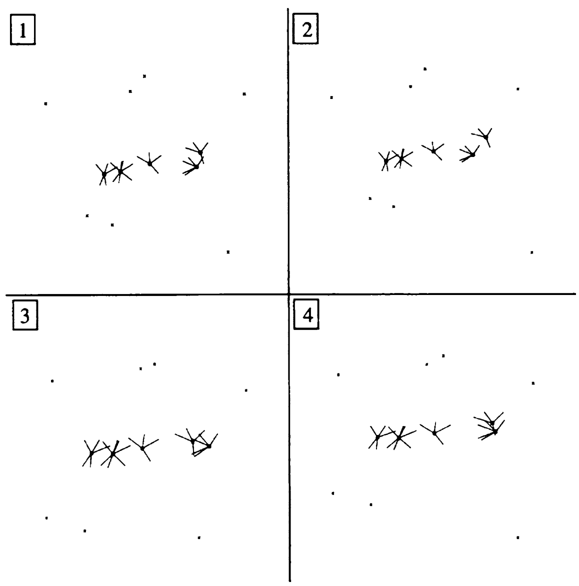

Figure 11. Four solutions to one minimal case of seven points in five images. Beacons are indicated by '+'.

In Table 6 a setting with five cameras and seven points is shown.

This is a minimal problem but not a prime one. The sub-problem of the first three cameras and the first five points is a prime problem which can be solved using the methods in [7]. This gives two solutions. The solution of the whole problem can then be found by the (2, 2) extension described in section 6.3. This gives two solutions for each of the original solutions, in total four solutions. In this case the solutions had all positive depths. The resulting solutions are shown in Figure 11.

8. Conclusions

In this paper we have begun to classify and solve structure and motion problems for calibrated 1D retina vision with missing data. We have introduced a notation on so called prime structure and motion problems. These are the problems that if solved will allow solutions to all structure and motion problems. Similar to the prime numbers there are infinitely many such prime problems.

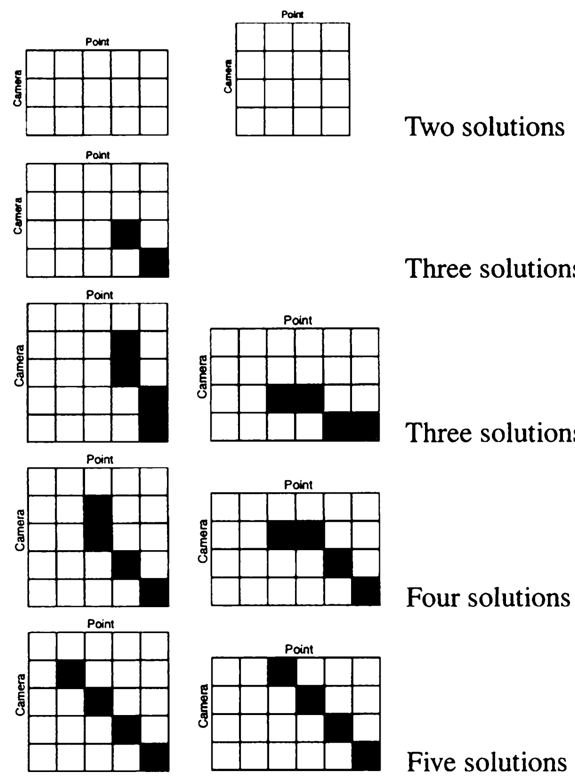

Figure 12. Solved prime problems.

In the paper we have given methods for calculating the number of such problems with a given size and also methods for finding representatives of each such problem. We have also begun our work on actually designing algorithms that solve the structure and motion problems for some of these instances. In Figure 12 all solved prime problems are shown.

More work is however needed in order to understand and solve these problems. We hope to be able to give more results in this direction in a near future.

Acknowledgments

This work has been supported by the Swedish Research Council (Vetenskapsrå det), project 221-2000-476.

References

1. M. Armstrong, A. Zisserman, and R. Hartley, "Self-calibration from image triplets," In *Proc. 4th European Conf. on Computer Vision, Cambridge, UK*, Springer-Verlag, pp. 3–16, 1996.
2. K. Åström, "Automatic mapmaking," In D. Charnley, editor, *Selected Papers from the 1st IFAC International Workshop on Intelligent Autonomous Vehicles, Southampton, UK*, Pergamon Press, pp. 181–186, 1993.
3. K. Åström, " *Invariancy Methods for Points, Curves and Surfaces in Computational Vision*," PhD thesis, Dept of Mathematics, Lund University, Sweden, 1996.
4. K. Åström, "One-dimensional retinae vision," In E. Bayro-Corrochano, editor, *Handbook of geometric computing, applications in pattern recognition, computer vision, neural computing and robotics*, Springer Verlag, pp. 209–304, 2005.
5. K. Åström, A. Heyden, F. Kahl, and M. Oskarsson, "Structure and motion from lines under affine projections," In *Proc. 7th Int. Conf. on Computer Vision, Kerkyra, Greece*, pp. 285–292, 1999.
6. K. Åström and F. Kahl, "Ambiguous configurations for the 1d structure and motion problem," *Journal of Mathematical Imaging and Vision*, Vol. 18, No. 2, pp. 191–203, 2003.
7. K. Åström and M. Oskarsson, "Solutions and ambiguities of the structure and motion problem for 1d retinal vision," *Journal of Mathematical Imaging and Vision*, Vol. 12, pp. 121–135, 2000.
8. S. Carlsson, " Duality of reconstruction and positioning from projective views. In *IEEE Workshop on Representation of Visual Scenes*, IEEE, pp. 85–92, 1995.
9. S. Carlsson and D. Weinshall, "Dual computation of projective shape and camera positions from multiple images," *Int. Journal of Computer Vision*, Vol. 27, No. 3, pp. 227–241, 1998.
10. D. Cox, J. Little, and D. O'Shea, " *Using Algebraic Geometry*," Springer Verlag, 1998.
11. O. D. Faugeras, L. Quan, and P. Sturm, "Self-calibration of a 1d projective camera and its application to the self-calibration of a 2d projective camera," In *Proc. 5th European Conf. on Computer Vision, Freiburg, Germany*, Springer-Verlag, pp. 36–52, 1998.
12. J.B. Fraleigh, " *A first course in abstract algebra*," Addison-Wesley, 5th edition, 1994.
13. R Gupta and R. I. Hartley, "Linear pushbroom cameras," *Pattern Analysis and Machine Intelligence*, Vol. 19,No. 9, pp. 963–975, sep 1997.
14. K. Hyyppä, "Optical navigation system using passive identical beacons," In *Proceedings intelligent autonomous systems, Amsterdam*, pp. 737–741, 1986.
15. C. B. Madsen, C. S. Andersen, and J. S. Srensen, "A robustness analysis of triangulation-based robot self-positioning," In *The 5th Symposium for Intelligent Robotics Systems, Stockholm, Sweden*, pp. 195–204, 1997.
16. M. Oskarsson, K. Åström, and N. C. Overgaard, "Classifying and solving minimal structure and motion problems with missing data," In *Proc. 8th Int. Conf. on Computer Vision, Vancouver, Canada*, pp. 628–634, 2001.
17. M. Oskarsson, K. Åström, and N. C. Overgaard, "Minimal cases of the structure and motion problem with missing data for one-dimensional retinae," In *Proc. 12th Scandinavian Conf. on Image Analysis, Bergen, Norway*, pp. 482–489, 2001.
18. L. Quan and T. Kanade, "Affine structure from line correspondences with uncalibrated affine cameras," *IEEE Trans. Pattern Analysis and Machine Intelligence*, Vol. 19, No. 8, pp. 834–845, August 1997.
19. C.C. Slama, editor, " *Manual of Photogrammetry*," American Society of Photogrammetry, Falls Church, VA, 4:th edition, 1984.
20. P.H.S. Torr and A. Zisserman, "Robust parameterization and computation of the trifocal tensor," *Image and Vision Computing*, Vol. 15, No. 8, pp. 591–605, 1997.
21. B. Triggs, "Matching constraints and the joint image," In *Proc. 5th Int. Conf. on Computer Vision, MIT, Boston, MA*, pp. 338–343, 1995.
22. W. Wei and J. Xu, "Cycle index of direct product of permutation groups and number of equivalence classes of subsets of z_ν," *Discrete Mathematics*, Vol. 123, pp. 179–188, 1993.

Magnus Oskarsson received his M.Sc. degree in Engineering Physics in 1997, and Ph.D. in Mathematics in 2002 from the University of Lund, Sweden. His thesis work was devoted to computer vision with applications for autonomous vehicles. He is currently an assistant professor at the Centre for Mathematical Sciences, Lund University, where his teachings include undergraduate and graduate courses in mathematics and image analysis. His main research interests are in algebra and optimization with applications in computer vision, cognitive vision, and image enhancement.

Kalle Åström received his B.Sc. in Mathematics in 1990, M.Sc. degree in Engineering Physics in 1991 and Ph.D. in Mathematics in 1996 from Lund University, Sweden. His thesis was awarded Best Nordic Ph.D. Thesis in pattern recognition and image analysis 1995-1996 at the Scandinavian Conference in Image Analysis, 1997. He has been a postdoctoral research fellow, associate professor and is now professor at the Centre for Mathematical Sciences, Lund University. His teachings include undergraduate and graduate courses in mathematics, image processing and computer vision. His current research interests include stochastic analysis of low level vision, computer vision for autonomous guided vehicles, geometry and algebra of multiple views of points, curves and surfaces, cognitive vision, handwriting recognition and medical image analysis.

Niels Chr. Overgaard completed his PhD degree in the area of mathematical analysis in 2002 at Lund University, Sweden. He received his MSc in mathematics and physics from Aalborg University, Denmark, in 1992. He is currently a Post Doc in the Applied Mathematics Group at Malmö University, Sweden, where he is involved in research on image analysis and mathematical biology. He has contributed 18 papers to international journals and conferences. His main research interests are PDEs, distribution theory, and calculus of variations, and their applications to medical image segmentation, active contours, computer vision problems, and biofilm modelling.